CESS-Cat dependency treebank for the CoNLL-2007 shared task

(Release 1.0 - January 2007)

Provider
CLiC Centre de Llenguatge i Computació
Universitat de Barcelona
http://clic.fil.ub.es
Contact person: Elisabet Comelles (Eli.Comelles@thera-clic.com)

With the collaboration of
NLP Research Group
Universitat Politècnica de Catalunya
http://www.lsi.upc.edu/~nlp
Contact person: Lluís Màrquez (lluim@lsi.upc.edu)

Files provided in this distribution:

Corpus:

catalan_cess-cat_train.conll	training corpus (430,844 words; 14,958 sentences)
catalan_cess-cat_test.conll	test corpus (5,016 words; 167 sentences)

Documentation:

README	this document
head.txt	table with heads
funct.txt	table with functions
POS-tagset.pdf	Catalan tagset, used to produce fine/coarse POS info.
cess-cat-conll.doc	licence for researchers (doc/rtf/text versions included)

The CESS-Cat dependency Treebank is created from the CESS-Cat Dependency Treebank, developed under the Spanish Research Project CESS-ECE (HUM2004-21127). More information and annotation style guides on that treebank can be found at the project home page: http://www.lsi.upc.edu/~mbertran/cess-ece. There, you can also find a demo interface for on-line querying the treebank.

The translation of the constituent trees into dependency trees (CoNLL-2007 shared task formatting) has been automatically done by using the information of two tables: HEADS (head.txt) and FUNCTIONS (funct.txt). These tables explain how to find the heads of the syntactic constituents and how to label relations with syntactic functions. They are briefly described below.

HEADS

The **format** of the head file (head.dat) is the following:

tag1 = (operator) tag2

where "tag1" is the mother and "tag2" is the daughter head.

The **Conventions** used in the head table are as follows:
"<": used to mark the beginning of a pos-tag.
"<>": used to mark the whole pos-tag.
"{": used to mark the beginning of a constituent.
Complete constituent tags are directly written.

Operators used in the head table are:
"rightmost": selects the rightmost element of a sequence of two or more.
"leftmost": selects the leftmost element of a sequence of two or more.
"only_one": works for the cases in which there is only one element of a given type.

Regular expressions:
When dealing with sentence coordination, a regular expression is used in order to select the same mother type with or without coordination, but always avoiding verbless sentences: {S(|.co)$

Examples:

1. espec.fp = <da
   The head of the "espec.fp" node is the element having a pos-tag starting with "da".
2. espec.fp = leftmost <di
   The head of the "espec.fp" node is the leftmost element having a pos-tag starting with "di".
3. espec.fp = <rg>
   The head of the "espec.fp" node is the element having "rg" as a pos-tag.
4. grup.nom.co = leftmost {grup.nom
   The head of the "grup.nom.co" node is the leftmost constituent whose tag starts with "grup.nom" (this includes "grup.nom.ms, grup.nom.fs, grup.nom.co", etc).
5. gv = rightmost <vsp
   The head of the "gv" node is the rightmost pos-tag starting with "vsp".
6. gv = only_one <v

The head of the "gv" node is the pos-tag starting with "v" if there is only one pos-tag starting with "v".

7. INC = sn

The head of the "INC" node is the constituent having "sn" as tag.

8. S.co = leftmost {S(|.co)$

The head of a coordinated sentence is the leftmost sentence or the leftmost coordinated sentence.

The **order** in the head table is a crucial element: when it is said that the first head of a given constituent is a given element, it means that at any time that this element appears, it is the head. The second head is only considered when there is no element of the first type for that given constituent.

Discontinuity

Discontinuity is dealt with in CESS-Cat in two different ways:

1. If there is a part of a constituent which is not at "its" natural place, there is an index (.1) both in the head and in the modifier. This index is a suffix of a constituent tag.
2. If there is a whole constituent which is not at "its" natural place, there is an index (.F or .NF) in the modifier, not in its head. In this second case, the tag suffix appears in the functional tag and its meaning is "(function) of the "next" finite clause (.F) or non-finite clause (.NF)".

The discontinuity always applies within the same sentence: whenever there is a "c" (*complement*) as the final element of a tag, there is an "n" (*nucli*)[1] as the final element of another tag.

Specifically:

".1n" stands for the core of a discontinuous constituent

".1c" stands for the complement of a discontinuous constituent

".Fn" stands for the core of a discontinuous function

".Fc" stands for the complement of a discontinuous function.

".NFn" stands for the core of a discontinuous function.

".NFc" stands for the complement of a discontinuous function.

There is one sentence in the Treebank which has two "NF" suffixes, so the second has the suffixes ".NF2n" and ".NF2c".

FUNCTIONS

Main functions, that is, functions related to the verb, are given in the Treebank. They are all in capitals and attached to the constituent node by means of a hyphen (-). Regarding the dependency conversion more functions are needed. We propose a function table (func.dat) to assign the remaining syntactical functions.

The **format** of the function file (funct.dat) is the following:

tag1 < tag2 = function tag

[1] *Nucli* is the Catalan word for English *core*

where “tag1” is the daughter, “tag2” is the mother and “function tag” is the function of the daughter with respect to its mother. If tag2 = *, this means that it applies for any mother (the * must be the only element in this position).

Regarding the rest of functions all the tags have been established (see funct.dat file). The criteria to establish the functional tags are:

1. Using the constituent tags of the daughter and the mother. For instance, ESPEC-SN means that there is a specifier (ESPEC) depending on a nominal phrase (SN).
   espec.fp.co < sn = ESPEC-SN
2. Simplifying as much as possible the tags in order to reduce the total number of tags. This simplification has been done in different ways:
   a) Removing all the dots in tags.
   b) Removing all the morphological information in tags.
   c) Generalizing as much as possible the functions; for instance, the INC node depends on different structures in the Treebank.

Special cases:

1. functional tag “CONJUNCT/ADJUNCT”: this means that the function can be either “CONJUNCT” or “ADJUNCT”. It's a CONJUNCT when, ignoring the first and the last daughter, any of the other daughters is a coordinating element (“coord”, “Fc” or “Fx"). Otherwise, it is an “ADJUNCT”.
2. functional tag “PUNC-CO/PUNC-SEP”: Commas (PoS = '**Fc**') and semicolons (PoS = '**Fx**'), which are daughters of a coordinated mother (ConstituentTag='***.co**'), will be given the separator function ('**PUNC-SEP**') if they are the first or the last daughter. They will be given the coordinating function (**PUNC-CO**') if they are in the middle.